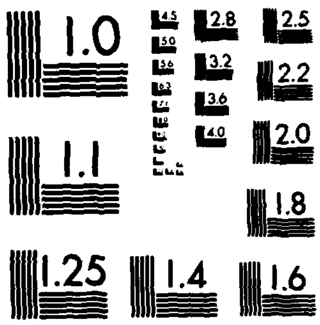
AD-A146 585 AN EXPERIMENT IN THE VALUE OF MILITARY INTELLIGENCE(U) 1/1
NAVAL POSTGRADUATE SCHOOL MONTEREY CA
K M O'BRYANT ET AL. MAR 84

UNCLASSIFIED F/G 15/4 NL



OPY RESOLUTION TEST CHART

(2)

NAVAL POSTGRADUATE SCHOOL
Monterey, California

AD-A146 585

THESIS

AN EXPERIMENT IN
THE VALUE OF MILITARY INTELLIGENCE

by

Kenneth Michael O'Bryant
Robert George Risney

March 1984

Thesis Advisor: F. R. Richards

Approved for public release, distribution unlimited

84 10 09 019

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER	2. GOVT ACCESSION NO. AD-A146 585	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) An Experiment in the Value of Military Intelligence		5. TYPE OF REPORT & PERIOD COVERED Master's Thesis March 1984
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Kenneth Michael O'Bryant Robert George Risney		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Postgraduate School Monterey, CA 93943		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Postgraduate School Monterey, CA 93943		12. REPORT DATE March 1984
		13. NUMBER OF PAGES 79
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release, distribution unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

chess
extra sum of squares
information overflow
Scheffe's analysis
value of intelligence
war game
intelligence

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

This thesis is an investigation into the value of intelligence on enemy position and strength during a simulated battle experience. An experiment was conducted to determine if there was an amount of intelligence which could statistically be shown to be optimal, with more or less intelligence resulting in a degradation in performance by the decision maker. A variation of chess was utilized as the basic war gaming model. Subjects were provided different levels of intelligence on the enemy's strength and position. A computerized chess game calculated all enemy moves. All aspects of the experiment

next page

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

S/N 0102-LF-014-6601

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered) including filtering of intelligence, communications between display terminals, and data collection were under software control.

The analysis of the data obtained from the experiment suggests that the amount of intelligence provided did correlate with player performance, and that there exists a level of information such that additional information leads to decreased performance.

Approved for public release, distribution unlimited

An Experiment in the Value of Military Intelligence

by

Kenneth Michael O'Bryant
Lieutenant Commander, United States Navy
B.S. in Marine Engineering, United States Naval Academy, 1971

Robert George Risney
Captain, United States Army
B.S., University of Nebraska, 1977

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN SYSTEMS TECHNOLOGY
(COMMAND, CONTROL, AND COMMUNICATIONS)

from the

NAVAL POSTGRADUATE SCHOOL
March 1984

Author: Kenneth Michael O'Bryant

Author: Robert G. Risney

Approved by: F R Richards
Thesis Advisor

Gary R Porter
Second Reader

Michael G. Sovereign
Chairman, Command, Control, and Communications Academic Group

K T Marshall
Academic Dean

ABSTRACT

This thesis is an investigation into the value of intelligence on enemy position and strength during a simulated battle experience. An experiment was conducted to determine if there was an amount of intelligence which could statistically be shown to be optimal, with more or less intelligence resulting in a degradation in performance by the decision maker. A variation of chess was ~~utilized~~ used as the basic war gaming model. Subjects were provided with different levels of intelligence on the enemy's strength and position. A computerized chess game calculated all enemy moves. All aspects of the experiment, including filtering of intelligence, communications between display terminals, and data collection were under software control.

~~The~~ analysis of the data obtained from the experiment suggests that the amount of intelligence provided did correlate with player performance, and that there exists a level of information such that additional information leads to decreased performance. Author-provided keywords [illegible]

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENTS

We wish to gratefully acknowledge the guidance and encouragement of our thesis advisor, Professor F. R. Richards. Without his expertise this thesis would not have been possible. Special appreciation is also due our second reader, CDR Gary Porter, USN, whose time, assistance, and class room support were invaluable. Additionally, we would like to thank Major W. Ted Farmer, USA, for his initial help in getting our experiment off the ground and his constructive input along the way. And finally, sincere appreciation and thanks to our wives, Rozella and Marlene, for their assistance, understanding, and moral support.

I. INTRODUCTION

A. BACKGROUND

Voluminous amounts of research have been conducted in the recent past concerning what information is used by a leader to make decisions on a tactical or strategic battlefield. Studies considering the value of intelligence to a decision maker make up a quite substantial proportion of this research. It was our hope that this thesis could add a little to the understanding of the importance to a force commander of military intelligence about the enemy.

Most of the research on the value of intelligence has been loosely structured and is subjective in nature. The tremendous scope that surrounds the whole idea of studying the "value of intelligence" seem to predicate a broad overview style of research rather than rigorously controlled scientific effort.

Our goal was to look at the value of intelligence in a quantitative way. To accomplish this, it was recognized that the magnitude of this study must be strictly confined in order that numerical results, rather than simply observations or personal impressions, could be obtained. Conclusions based on real experimental data would be sought, not generalized opinions or observations.

There are a number of ways to gather data to study the value of intelligence to a leader. Past research has used everything from historical reports on actual battles or wars, to results of training exercises and operations. The war game is a vehicle becoming more in vogue to generate useful data for analysis of this kind. Because of our desire for a strictly controlled environment for the thesis's investigation into the value of intelligence, a war game seemed to be a desirable medium to use. An experiment conducted using a "credible" war game, we believed would provide the definitive conclusions necessary to numerically justify the inferences expected to be made on the value of intelligence.

War games can be classified in a multitude of fashions, such as purpose of the game, scope or level of the game, type of simulation or model, method of evaluation, and level of abstraction. The type of war game we sought would be classified in the JCS Joint War Gaming Manual [Ref. 1] as a research type war game. Our need for the game was for use as a testing vehicle for research into the value of intelligence.

Current computerized war games range from very large analytical simulations which take hours to calculate one game turn, to small educational war games designed to provide the players with semi-realistic decision-making experience. These types of games are designed to train individuals, not provide analytical data for experimentation purposes. Most lack the crucial ability to provide computer generated decisions. War games such as Naval Warfare Interactive Simulation System (NWISS) and the McClintic Theater Model (MTM) rely on the human to make the decisions. The computer simply keeps track of the multitude of parameters concerning the current game situation, and updates those parameters, given the decisions made by the players. We sought a one-sided game where the computer could provide quality, consistent decisions over a number of game turns.

These above requirements lead us to the selection of a variation of chess as the principal war game for the experiment. The reasons behind selecting chess over more complex war games will be discussed in greater detail in the experimental design chapter. It will suffice here to say that chess met the requirement of being simple enough to allow us to manipulate, automate, and analyze it, while still maintaining what we and many others believe to be a reasonable surrogate of a battle experience.

B. PURPOSE

The purpose of this thesis was to design, conduct, and analyze an experiment which would allow us to study the value of intelligence in a low-level war gaming situation. The experiment was to confront military subjects with a variety of different amounts of information about "friendly" (white) or "enemy" (black) positions on a chess board. Data would be collected from these confrontations (trials) and from the analysis of this data it was hoped that some definitive results as to the "value of intelligence" could be obtained.

A number of different hypotheses are possible when studying the value of intelligence. Intuition told us that it was probably true that the more information given, the better a commander would perform, as long as the information provided was relevant to the situation, and the quantity of the information was controlled to avoid "information overflow". It was our desire to prove or disprove this intuition, by studying information as it applies to intelligence about the enemy.

Therefore, our primary hypotheses for the experiment were:

1. The amount of intelligence provided to a decision maker (in this case a chess player) on enemy positions and strength was positively correlated with the player's performance after using the information provided.

2. There was a specific amount of intelligence which could be shown to be optimal, and if more or less than the optimal amount of intelligence was provided, the performance of the player would be degraded.

There were other related secondary hypotheses or experimental issues which we felt naturally arose from the testing of the primary hypotheses. One of the most obvious issues that must be considered was that it is probably true that experienced commanders performed better. In our experiment this would relate to the better the chess player, the better the performance, regardless of the amount of intelligence provided. Another hypothesis would be that additional intelligence might be of more use to the weaker, less experienced decision maker. A strong chess player might not use or even want information that a weaker player possibly would find extremely useful.

And finally as a postulate to these hypotheses, we hoped to show that one of the most basic of war games (chess) could be used to formulate definitive results which could meaningfully contribute to overall understanding of the research area. Although ancient in design and considered rudimentary in scope by some war gamers, the fundamental ideas of position, strength, and movement in chess could be naturally related to the same decision-making parameters one must take into account to make acceptable decisions on the modern battlefield. There are of course differences in rules, timing, scope, and magnitude, but the underlying principles are the same.

C. APPROACH

The approach taken consisteo of four distinct phases, each of which will be reported on in a separate chapter of this thesis. In general, the four actions were:

1. Formulate the hypotheses we wished to test and design an experiment which would allow us to accurately evaluate these hypotheses. Chapter II reports on areas such as selecting the appropriate war game, devising a credible measure of effectiveness, and developing a suitable mathematical model which would support the experimental aims. The criteria involved in selecting the necessary subjects for the experiments, along with equipment requirements, are also discussed.

2. Design the software needed to support the experiment. In Chapter III the computer programs developed for the experiment are examined. We did not feel it necessary to include the multitude of code written for the experiment, in the thesis. We felt it more appropriate that each of the sections, along with some of the modules contained in the sections, be discussed in a more holistic manner within the body of the report. This should lead to a better understanding of just what was required of the software and how it fulfilled those requirements.

3. Conduct the experiment. Chapter IV recounts the particulars on the actual execution of the experiment. The procedures involved in administering and controlling the experiment are explained, together with more detailed information on the players and equipment used.

4. Analyze the results and draw our conclusions. Chapter V explains the approach and the specific statistical tools used to reduce the data. The conclusions we reached are presented in Chapter VI.

II. EXPERIMENTAL DESIGN

A. SUBJECTS

The desired subject for our experiment was an adult with some military experience that could be brought to bear in making decisions of a tactical nature. The particular service connection would be unimportant because the game chosen would not favor any particular military background. The subject would preferably have some familiarity with the game of chess but would not be a highly experienced player. Also, the subject should be familiar with the use of computer terminals in general as a communications device to avoid possible contamination of the results of the experiment from computer angst.

The other driving factor was that a sufficiently large group of subjects was required to obtain enough data points for analysis. Exactly how many were enough could not be determined at the outset because the sample size would, of course, depend on the other design factors and the amount of data scatter actually observed. After completing the initial design of the experiment, pilot trials were conducted to get some idea of the variation in scores we could expect and to refine some of the experiment parameters. Based on those trial runs we felt that "the more the better" was the answer, expecting that the number of volunteers we could enlist or conscript would be smaller than the number physically possible to process in the time we were allotted for priority use of the WAR Lab facilities and that we would probably observe significant variations in the scores.

As things turned out, we had thirty-one subjects available meeting our criteria, and the processing of their trials and data took essentially the entire time available. Time played a part in other design parameters, also, as will be discussed later. For more specific information on the subjects actually involved, see Chapter IV, Conduct of the Experiment.

B. APPARATUS

The experiment was run in the Wargaming Analysis and Research Laboratory (WAR Lab) at the Naval Postgraduate School. A computer capable of handling the required interfacing, multi-terminal coordination, visual display, and data collection was essential. An atmosphere in which terminals and work space could be reserved for experimental use and in which the subjects and umpires could be relatively undisturbed was equally important. The WAR Lab offered those advantages.

All artificial intelligence in the process of deciding Black's moves was provided by the "Super-Nine Chess Challenger" (a commercial chess game manufactured by Fidelity Electronics, Ltd.).

The VAX-11/780 mini-computer in the WAR Lab was also used with a separate chess game program to allow the subjects to play practice games for nomenclature and chess familiarization.

More detail about the specific equipment used can be found in Chapter IV, Conduct of the Experiment.

C. PROCEDURES

The first step was the selection of the hypothesis and a suitable vehicle with which to test it. We believed that the amount of information a decision maker has and the results of his decision are correlated. We thought this to be particularly true in the field of military intelligence and combat. We expected that there would be an optimum level of intelligence; too little intelligence resulting in too much uncertainty and too much resulting in an overload of the information assimilation process which could disguise key issues in the flood of minutia. To test our hypothesis we needed a test bed of some sort. To improve credibility of the results an actual combat situation would have been the best test bed but that is obviously not practical. It would also not allow for duplication and would be virtually impossible to control rigidly. For the same reasons a field exercise, probably the next most credible format, was not practical, either. War games are generally accepted as the next echelon of credibility for military situations and can be run economically, repeatedly, and with varying degrees of control. Ideally, the war game selected would have some easily arrived at measure of effectiveness (MOE), be of relatively short duration, require only one subject at a time, not be so closely allied to one area of combat as to give significant advantage or disadvantage to subjects with any specific military experience, and yet would still be a suitable surrogate for combat to be of value for demonstration. A critical consideration in running an experiment is the ability to control and account for the factors that may influence its outcome. Unfortunately, realism of the battlefield environment and rigid experimental control are diametrically opposed conditions. In order to keep our experiment as simple as possible and to be able to extract statistical data for hard analysis we opted for tight experimental control and sacrificed battlefield realism.

A variation of chess was chosen because it most nearly met all the criteria. Chess originated as a war game and is the oldest surviving one in the western world. In the nineteenth century the German Army's General Staff used a variation of chess called "Kriegspiel" ("war game") as a training aid in tactical and strategic thinking. In "Kriegspiel," two opponents play chess but each sees only his own pieces on his board; an umpire provides the necessary interface between the two players. The object is still to destroy the enemy by capturing the enemy's king, but the process is much more difficult. Knowledge of the opponent's strength and position must be derived from scouting, losses, engagements, etc.

The game that we used in our experiment is another variation of chess in which the amount of intelligence provided can be controlled by software. In a normal chess game two opposing players match wits developing strategy and counter-strategy until one is beaten. As with any other human endeavor, the skill with which one plays varies from game to game. This is true even for the greatest chess masters; certainly it would be true had we used one of our umpires to always play the opposition. Having our subjects play against each other was an even less viable solution. That would have required a larger number of trials to get sufficient data, would have required much more of each subject's time, and would have provided very inconsistent opposition. We felt it was important to provide a consistent opposition for our subjects in order to remove the possible confounding of two subjects' relative chess acumen and to avoid the necessity of a very large number of trials. Therefore, all subjects played against the same computerized chess game at the same level of play. We chose to set the Chess Challenger at its lowest (easiest) level for a number of reasons. First, how well the Chess Challenger did was not important as long as neither white nor black frequently decimated the other. We expected that most of our subjects would be novice chess players so that even in a normal game the artificial intelligence in the Chess Challenger would be a significant challenge. How well the Chess Challenger scored was far less important to the experiment than the fact that its play was always at the same level. Therefore, while a higher level might have been more of a challenge to an advanced player, it was more important to use a level that would be easy enough to give our novices a chance at avoiding early checkmate. Pilot trials indicated that the lowest level of play on the Chess Challenger would provide an adequate challenge. Our decision was proven correct in that even against the lowest level setting none of our subjects were ahead at the evaluation point. Lastly, to minimize the minimum time required for each move we wanted the reaction of the Chess Challenger to be as quick as possible. Using its lowest level meant that its decision tree analysis was kept simple with a resultant decision time of approximately five seconds. The computerized opponent always had perfect information (the normal view of the chess board with all active pieces).

The second step was selection of a suitable MOE. This is often a difficult problem. Consider the example of trying to evaluate the effectiveness of a new anti-aircraft system protecting a critical target. Possible MOE's are the number of bombers shot down per 1000 rounds fired or the amount of damage suffered from bombing before and after installation of the new system. Which is a better MOE depends on what one really means by the "effectiveness" of the new system. Correct choice of an MOE requires careful scrutiny of the underlying questions one is trying to answer and accurate translation of the requirement into measurable quantities. At this stage we made a general decision to use a point value system based on material strength, board positions, and mobility. Points would be determined for the initial board and then again at some number of moves later; the algebraic difference in the scores, corrected for any penalties became the MOE. The specific method of doing this and assigning penalties was worked out after the other details of the experiment were determined and is described in Chapter V.

The next major step was to develop a mathematical model of the experiment that would account for as many parameters and influences as possible. To allow analysis for differences in chess playing expertise each subject was asked at the beginning of each trial run to indicate into which of four categories the subject fit: novice, some experience, frequent player, or tournament player. To remove foreknowledge of the board setup we started each trial in mid-game. To minimize the possible "learning" of initial piece dispositions, four separate mid-game start points were constructed [Figures II-1, II-2, II-3, and II-4]. We felt this was important because otherwise a player with any experience at all would start out with "perfect information" about the opposition regardless of what information was displayed. In such a case a player's experience as a chess player would take on even more significance due to the expert's ability to extrapolate probable black counters to his moves, knowledge of opening game strategy, etc. Using a mid-game start point also offered the advantages of avoiding end-game strategy (in most cases), of clearing the board somewhat to facilitate greater movement, and of eliminating most of the uninteresting early swapping of pawns. These positions came from playing the first eight to ten moves of tournament games discussed in the chess column of the local newspaper and stopping at a point of approximately even strength. The sequence in which the subjects faced the initial setups was varied systematically so as to appear random to the subject but to yield approximately uniformly distributed sequences of play. For example, the number of subjects playing with intelligence level one against board setup one on their first game was approximately the same as the number facing intelligence level three and board setup two on their first game, etc. The order in which different information levels were used was similarly varied.[1] By collecting "trial number" as one of the experimental parameters, it was possible to analyze the effect, if any, of learning. By using the four setups to eliminate one confounding factor, we introduced another, the effect of playing one setup versus any other one. Therefore, the initial game setup was always recorded for analysis of its effect. The final item to go into our experimental design and the major factor of interest was the intelligence level provided to a subject during a game. We devised six different levels of information (which are explained in detail in Chapter IV) but found that it was not feasible to use them all. The six levels came about by determining what specific types of information could be provided and how those types could be combined. Practical limitations on the time allotted to run the experiment and use the facilities, on the amount of game time each subject was willing to provide, and on the number of subjects available forced the reduction to some smaller number. On the the basis of trial runs conducted with ourselves as subjects we felt four

---

[1]Table V-1, the data file from the experiment, shows the sequence of play. One of the umpire's actions before the game was to use a copy of the uncompleted data file as the guide for selecting the appropriate intelligence level and board setup for each subject on each trial. There is no significance to the assignment of subject identification number.

levels were the maximum that could be used. That was a compromise. The four selected were chosen on the basis of those trial runs so that we had a good spread of game types (intelligence levels); choosing the three lowest levels plus one more, for example, would have probably produced only a small or nonexistent spread of scores because all three allowed very little information to the subject. Intuition and our trial runs showed the four levels selected held the best promise for delivering meaningful data points.

Combining the factors discussed above, we can represent our experimental design model as:

$$Y = a_o + \beta_i + \gamma_j + \alpha_k + \rho_l + e$$

where:

Y = MOE

a_o = some base level

β_i = information levels

γ_j = trial number

α_k = initial board setup

ρ_l = subject's experience level

e = unknown or uncontrolled error.

Other factors were considered but not represented in the model. As a surrogate for the operational pace of combat decision making, the subject was allowed two minutes per move without penalty. Two minutes was arbitrarily chosen after pilot trial experience showed it to be adequate. Practical constraints were also a factor. Allowing two minutes per move provided adequate time for each subject to play four games during the three hours each was available. The subjects were told that there was a penalty for exceeding two minutes per move but were not told the exact nature of the penalty. We did that to force the pace of the game without introducing the question of intentionally trading a known penalty for additional decision time. Any penalty was assessed after play stopped at the rate of one pawn's material value (256 points) for each one minute or fraction thereof of cumulative time over two minutes per move for the ten moves of each trial. However, to allow the player to study the initial conditions, the first move was not penalized. The number of moves per trial until evaluation was set at ten based on pilot trials to determine a suitable number. Too few moves would not allow time for the various factors to effect the score; too many moves would cause a large number of subjects to be checkmated resulting in a skewing of the scores. We felt that the alphanumeric board and move representation on the computer terminal would help mitigate the expected differences in the subjects' chess expertise. A similar chess game using the same representation and nomenclature was made available to all the subjects for practice. To prevent the introduction of another element into the experiment, outside aids such as a conventional chess board were not allowed.

III. SOFTWARE DESIGN

The design and development of the software needed to support the experiment took place over a 2 1/2 month period from July to September 1983. The final software product consists of 25 subroutines, presently located in 3 files on the VAX 11/780 computer in the WAR Lab. The files are open for public review in the .THESIS subdirectory of the CHESS directory, under filenames UMPIRE.FOR, PLAYER.FOR, and COMMON.FOR.

All programming was done in standard VAX-11 FORTRAN 77 [Ref. 2], with the exception of the inter-process communications via a systems mailbox. That code was written in FORTRAN formatted VAX 11/780 Systems Programming Calls. [Ref. 3]

The code is of course compatible with any Digital Equipment Corporation computer system running under a VAX/VMS operating system. With the possible exception of the inter-process communications mentioned above, and some specific intrinsic function calls not supported in standard American National Standard FORTRAN-77, the program should compile and run on any computer system which has a FORTRAN-77 compiler.

A. COMPUTERIZED CHESS

To design a program which will intelligently play a game of chess is a tremendous undertaking. It was decided early in the formulation of the experiment that to actually write a program which could play chess with even a small amount of skill was not only far beyond the scope of this experiment but also unnecessary, since extremely competent chess programs exist and could be utilized easily.

A great deal of software was still needed to control the play of the game. This software was to be designed in accordance with established computerized chess principles. Although no actual artificial intelligence would be programmed to determine moves, all other portions of the game required software support. In addition, a major programming effort was required to construct the additional masks and screens necessary to control the amount of information given to the subject during the actual experiment. Since it had been decided that there would be little or no direct personal interaction between the subject and the umpire, all the communications which was to take place between the umpire and player, along with the amount of intelligence provided the player, required FORTRAN coding.

The basic ideas involved with the representation of chess in a computer are really quite simple. A chess board consists of 64 squares, organized in a square 8 X 8 matrix with a single item (the playing piece) possibly sitting on top of each of the squares. These pieces move around on top of the board according to specific rules which govern each type of piece. There is an object to the game, i.e. capture the opponent's king, and many general rules, such as pieces can not move off the board, which control overall play.

Currently there are two generally accepted methods used by computerized chess designers to represent a chess game inside a computer. One is the Shannon method, which will be described in much greater detail below. This method uses numerical arrays to represent the board, and utilizes a large set of procedures which emulate the general and specific rules of the game. The other method of representation is known as the "bit board" representation and utilizes a series of 64 bit words to portray the basic board, and all of the rules associated with moves from any position, using any piece, on the board. [Ref. 4] These "bit boards" enable the processor to do simple boolean logic operations such as 'AND' and 'OR' on combinations of the 64 bit words to generate rules. Since fetches from memory and logical operations are much faster than long procedures, this method reduces processor time, which is a most critical commodity if the chess game is to formulate computer generated moves.

For this experiment though, speed of computing was not a factor because the software did not generate chess moves. The Shannon method of game board representation was therefore chosen for the foundation of the chess program's design because of its overall simplicity and also the ease with which it can be programmed. Shannon suggested [Ref. 5] that each square on the chess board be looked upon as a "mailbox" which certain attributes, for instance whether the square has a piece on it or not, are stored. His original idea was to have sixty-four such "mailboxes" for the sixty-four squares on the board. More recent programs modified this representation to include hypothetical squares which are off the board. [Ref. 6] Our internal board representation took this updated approach and consisted of a one hundred twenty element array which could be though of as a 10 X 12 square board. [Figure III-1] Each mailbox could contain either zero, a positive or negative number between one and six, or the number ninety-nine. These numbers would tell the status of each square at any particular game time. For instance: a 0 meant the square was empty, a +1 meant there was currently a white pawn at that square, a -4 meant the square was occupied by a black rook, and the number 99 depicted a square which was off the playing board. [Table III-1] Using this system of off-the-board squares, the edges of the board could be easily detected. Although the necessity of these squares is not initially obvious, the use of these off-the-board squares should become clear when an example of a move is explained in the LMC subsection.

B. OVERALL STRUCTURE

Ignoring the differences between the umpire and player programs for the time being, the overall structure of the software consisted of seven different sections, each of which was designed to call various subroutines at different times during program execution. These sections are described below.

1. Introduction and Initialization

This portion of the code did the start-up and initialization chores, queried the user for primary experimental data entries, initialized the default playing board (or, if the user desired, set up a board to the player's specifications), and set up the timer used to time the length of each move.

2. Parser

The parser section converted the user's typed in move to the internal representation of the move. Depending on whether the umpire or player entered the move, the move would be entered in either the basic chess movement scheme (i.e. P/KB2 -KB3) or in the Chess Challenger's board portrayal (i.e. P/F2 - F3). This move would then be converted into the internal square and piece designations. Using the above move as an example, P/KB2 -KB3 would be translated into, if it is white's move at the time, pawn (+1 for white) at internal square 37 is to be moved to square 47.

The parser checked for illegal entries, and if an illegal move is made, the parser informs the user and continues to ask for moves until a legal entry (not necessarily a legal move) was made.

3. Legal Move Checker (LMC)

This portion of the program was one of the most complex and required a great deal of design effort and debugging time to get working properly. The idea behind the LMC was to determine whether an attempted move was legal, given that the LMC knew the origin and destination square, and the type of piece which the user wanted to move. Legal moves can then be determined by noting the mathematical relationships between squares.

For example, if a white knight was to be moved from its default starting position at QN1 to QR3, the possible legal moves can then be calculated by adding the following offsets to the origin square.

Origin Square = 23: +8, +19, +21, -8, -19, -21, & -12

Each of these squares are then matched against the destination square. If a match occurred, and the square is not occupied by a friendly piece or located off the board, it is a legal move. Figure III-2 shows how the default board would be internally represented at the start of a game. Using Figure III-1 as a guide, the above example shows that adding +19 and +21 are the only two legal moves from square 23. These two squares (42 and 44) are the only 2 squares which have a 0 or negative number in them. All other offsets contain either positive numbers (it is illegal to move on top of one of your own pieces) or a 99 (which means the attempted move is off the board). Since the desired move is QR3 (square 42) this then is a legal move.

This is a highly simplified example of how the LMC works. The moves for sliding pieces such as rooks or bishops are much more complicated to check, but follow somewhat the same principle using offsets and comparisons. The LMC looks only at regular moves and capture moves. A bit more of a streamlined approach could have been taken to eliminate some of the redundant offset additions; however, some of the same code was to be used in other sections of the program. Therefore some efficiency was sacrificed for clarity and generality. A more detailed explanation of this section can be found in the comments of the program's source code, or in an informative book on computerized chess which was used extensively in modeling the movement portion of the chess program. [Ref. 7]

4. Display

The display section was responsible for all output to either the terminal, the line printer, or separate files. It consisted of subroutines or modules which performed the following functions:

a. Display the board after each legal or illegal move. The internal representation of the board had to be converted into a representation suitable for display on the output device. The board display type chosen consisted of eight lines of 2 symbol groups which were either dashes or asterisks for white or black squares which were empty, or two letters to depict a piece that occupies a square. [Figure III-3] A black king for example would be BK and a white knight would be displayed as WN.

b. Decide how much information should be removed or added to the normal board display. This code, coupled with the intelligence determination modules of section 5 below, insured that the proper amount of intelligence was displayed to the subject, given the game scenario being run at the time.

c. Display a safe board to the subject. During different scenarios this board would be displayed to the subject using eight lines of three letter groups which informed the subject which squares on the board were safe from attack. [Figure III-4] The letter groups were in standard chess square terminology (i.e. KB7 = the square king bishop 7) and would be displayed only if that square was safe from attack.

d. Output to a separate file each move and the time it took to make the move. The program was designed to output each move to insure that if any data was lost, the game could easily be reconstructed. The standard chess move format was chosen for output; therefore, the umpire's moves required translation from Chess Challenger format before they could be written to the file.

e. Collect and save data points. Specifically at turn ten of each game, and additionally at any time the umpire chose, the program would query the umpire for an evaluation of the board situation at that time. This information, along with the actual board position, and all of the other experimental independent variables, would then be saved in the data file for future analysis.

f. Display to the other player the move that was entered. Since the umpire and subject played the game using different game board representations, each move required translation into the other player's format before it could be displayed on the opponent's terminal.

5. <u>Intelligence</u>

This section modelled the heart of the actual experiment since its functions were to derive the information on attackable pieces, pieces that were safe from attack, and squares that would be safe if a piece was to move there. The primary subroutines of this section were WHITE & BLACK-ATTACK and SAFE-BOARD. These subroutines in turn called many of the same routines used in the legal move checker, however normally with different input parameters and common variables. The basic idea of any of these routines was to check every square on the board for possible legal moves from that square, depending on whether pieces that could be attacked, or safe squares, were sought at the time. These possible legal moves would then be matched against the playing pieces relevant to the intelligence needed and a board would be constructed which would simply contain yes or no to the question of whether the square was, let's say safe from attack. These "boards" were just arrays of boolean variables which could then easily be matched one for one with the actual game board to display the proper information for the scenario being played at that time.

6. Castling

Castling, because of the many rules involved in this maneuver, was handled separately from the rest of the movement sections. Before a castle could be made, numerous rules had to be checked that were different from a normal move's rules. Also the move required the relocation of two pieces rather than the usual one. The parser would identify a request for a castle and then the following rules had to be checked before the move could be made.

a. Are the king and rook in their proper positions?
b. Has the king or rook you wish to move, moved before?
c. Are there any pieces between the king and rook?
d. Is the king in check?
e. Will the king move into or through check during the castle?

If all of these questions are resolved satisfactorily the castle would take place as requested. Otherwise, an illegal move message would be displayed on the terminal and the player would be asked for a different move.

7. Exchanging of Pawns

This section dealt with the situation which occurs when a pawn reaches row eight of the game board. If this situation occurred during the movement of a pawn, the code would query the user for the type of piece to be exchanged for the pawn, and then make the substitution as required.

C. MAIN PROGRAM

The above sections were integrated with a main program (either player or umpire) to form the executable program module. The subroutine integration was done primarily at linkage edit time since some modules were used in both the player and umpire processes. In addition to the seven major sections, there were a few other minor segments, such as a routine which determines whether a piece is in check, and a portion of code which would check to see if the player was still in check after a move was made.

The main program is a large repetitive loop. The flow of control would normally go through sections b, c, d, and e on each move [Figure III-4]. Section a would be executed only upon program start-up or reset. Sections f and g were executed only on demand.

D. DIFFERENCES BETWEEN UMPIRE AND PLAYER PROGRAMS

The overall structure of the umpire and player programs are generally the same. Each program is designed to run as a separate process. All subroutines which were common to both the umpire and player programs were compiled as separate routines and linked into each program separately. There are some basic differences between the two processes though, and they required individual program code.

1. All data collection and output was handled completely by the umpire terminal. The player had no control over the data that was saved.

2. The umpire would always see the highest level of intelligence at his/her terminal. The player would see only what was selected for the player to see by the umpire.

3. All timing was conducted only in the umpire program.[1]

4. The umpire process handled all the translations between move formats. These translations included: Chess Challenger to regular chess, regular chess to Chess Challenger, and perspective changes such as a move in the black's perspective translated to the same move in white's perspective. A black to white translation meant, for instance, that if black was going to move a piece from his KR3 to KR4, white would be told the move from his perspective, i.e. KR6 to KR5.

E. COMMUNICATIONS BETWEEN TERMINALS

All communications between the umpire and player were handled by the creation and use of a systems "mailbox". This mailbox acted as a buffer between the two processes. The two programs passed information to and from the mailbox using the VAX 11/780 systems input/output (queued) routines.

The mailbox size was 600 bytes. Although quite large, the majority of the time the only information passed through the mailbox was the actual 20 byte move entered by the player or umpire. The large size was necessary to

---

[1] This did not present a problem, even though it was the player's move that was being timed, because the actual time span being measured was from the time the umpire's move was sent to the player, to the time a legal move was received back from the player.

pass the initial board set up, entered by the umpire, to the player process before play could begin.

Each of the two processes was responsible for determining whether a move was legal or not, and then carrying out the actions required by the decision. Therefore, if the player entered an illegal move, the umpire terminal would receive the move and determine its illegality, just as would the player terminal. The system was designed in this matter to help the umpire control the flow of the game, even though it is definitely redundant in nature.

F. VARIABLES

Both local and global variables are used in the overwhelming majority of the software modules. Because of the nature of the program, and its need for an extensive amount of variables passed between procedures, common blocked variables were chosen over large parameter lists. Each common block was tailored for a specific use so that the number of global variables required in each subroutine could be kept to the minimum needed to perform its necessary functions.

G. COMMENTS

A final word on the structure of the comments and other documentation added to the program. At the beginning of each subroutine is a description of each variable local to that specific module, and each input or output parameter of that subroutine. At the beginning of each of the two main processes are descriptions of all global variables common to any or all the procedures of the process. Comments are interspersed throughout all the software. We tried to comment blocks of code as much as possible, rather than individual lines, to help in identifying program structure and enhance the readability of the code.

IV. CONDUCT OF THE EXPERIMENT

At the earliest stages of formulating this experiment an agreement was reached between ourselves and CDR Gary Porter, the instructor of the fall class of OS-4602, C^3 Systems Evaluation, to utilize the students in his class as subjects for this experiment. In exchange for the use of his class and classroom time, we would allow the experiment and its results to be used in class as a learning tool for teaching experimental design. Therefore, the time frame to conduct the experiment had to be convenient for both CDR Porter's class objectives and this thesis' requirements and goals. The time period agreed upon for execution of the experiment was a two week period in early October 1983. The actual experiment took a week longer than expected, lasting from 10-28 October 1983. The extra week was needed due to the determination, as the experiment proceeded, that some additional data points would be required for data analysis. Additionally, there was a significant number of the subjects that were unfortunately scheduled to be absent during a large portion of the initial two weeks, and there was not enough time in the remaining days to run these students through the experiment before they left.

The entire experiment took place in the WAR Lab of the Naval Postgraduate School (NPS), Monterey, California, located on the first floor in Ingersoll Hall.

A. PLAYERS

Overall there were 31 individuals who took part in the experiment. All students were military officers, with rank ranging from a Lieutenant

Colonel/05 to Lieutenant/03. All services were represented with 14 officers from the Navy, 7 from the Army, 8 from the Air Force, and 2 from the Marine Corps. Experience levels in playing chess will be discussed in more detail in the analysis chapter. It will suffice here to say that the experience level of the subjects as a whole was fairly low, with the most experienced palyer being unranked in the US Chess Federation and classifying himself as no more than an infrequent player. There were also a few subjects who had never played the game before they attempted the practice sessions which were scheduled a week before the actual experiment.

All but two of the individuals were part of either the Command, Control and Communications (C^3) or Space Systems Operations Curriculum. Strategic decision making experience of the group was low, as would be expected with officers of the above rank. Tactical decision-making experience, on the other hand, was much more prevalent, with many subjects having extensive ground or naval tactical warfare training and/or experience.

B. UMPIRE

Along with the authors of this thesis, two other students of the C^3 curriculum were used as umpires to control the experiment.[1] The umpire's job consisted of: preparing the two terminals and the Chess Challenger for playing the correct scenario at the appropriate intelligence level, giving the pre-experiment briefing to the subjects, providing the interface between the experimental computer program and the Chess Challenger, informing the

[1]These two umpires were also subjects, but acted as players before learning the umpires' duties to insure their data points were not contaminated by the additional information given to them on the experiment.

subject of the time left to make a legal move, and controlling the data collection aspects of the experiment.

C. EQUIPMENT

The equipment utilized to conduct the experiment consisted of: WAR Lab's VAX 11/780 mini-computer, two Digital Equipment Corporation VT-100/-102 video display terminals and keyboards [Figure IV-1], and a Chess Challenger computerized chess game. [Figure IV-2] One VT-100 terminal was used by the player and the other by the umpire. During the experiment, these terminals were controlled with the software program described in Chapter III. The umpire operated the Chess Challenger.

1. About the Chess Challenger

The Chess Challenger is a computerized chess game manufactured by Fidelity Electronics, Ltd of Miami, Florida. This game supplied the primary artificial intelligence tool utilized to figure all black's moves. It also was used to provide the evaluation function utilized in computing the player's relative board strength at specific times during play of the game. Additionally, the built-in timer of the Chess Challenger was used to keep track of the time left before the player was required to make the next move.[1]

---

[1]This timing was for umpire and player information only. Timing for penalty assessment was accomplished by the software program running on the VAX 11-780. Software controlled times however could not be displayed at the terminals without seriously interfering with the game boards presently displayed on the VT-100's.

The Chess Challenger is an extremely powerful chess game capable of playing at anywhere from novice to tournament level chess. It has been ranked by the US Chess Federation at approximately 1825 - 1850.[1] Although capable of playing at an extremely high level, the very lowest level was chosen for this experiment. The average response time for the game to make a move at this level was 5 seconds.

D. PROCEDURES

1. Physical Lay-out

The equipment listed above was set-up in an isolated corner of bay 3 of the WAR Lab during the execution of the experiment. The terminals faced each other with a 6' X 6' partition separating the player and umpire stations. [Figure IV-3] Partitions surrounded the player's working area to completely isolate them from outside interference in the lab. No distractions such as clocks, other terminals, or printers were in view or ear-shot of the subject.

2. Practice Session

While designing the experiment it was determined that there was a solid need for some training and/or familiarization in playing chess before the actual experiment could take place. Therefore, practice sessions on the computer were scheduled the week before the experiment started. Each subject was asked to log onto a terminal and play a chess game, similar to the experiment, for at least one hour. The practice game board display used

---

[1]This rank equates to a Class A player. Rankings are as follows: Grandmaster - 2600 and above, Senior Master - 2400 to 2599, Master - 2200 to 2399, Expert - 2000 to 2199, Class A - 1800 to 1999, Class B - 1600 to 1799, and Class C - Below 1600 points. [Ref. 8]

the same type of symbology board as the experiment. The way a move is entered was also identical. The practice chess game was played against the computer, which used a chess program different from that used in the actual trials.[1]

3. Scheduling

As described in the experimental design chapter, each subject was required to play the game four times. At the beginning of the experiment each individual was asked to sign up for three hours of time to play the four games. Depending on how fast the players made their moves, the four games would last anywhere from two to three hours. To try to avoid boredom and fatigue, the subjects were encouraged to sign up for three non-contiguous hours of play.

4. Actions Before Each Game

Before the start of each game a series of actions were required to be accomplished.

a. The umpire initially would reset the program and make the selection as to how much intelligence the player would be allowed to have during this game. A menu would appear on the umpire terminal listing six options, corresponding to six different levels of intelligence to be presented to white. [Figure IV-4] The umpire would choose the option

---

[1]Although used for practice, this chess game was found to be entirely unsuitable for determining moves in the actual experiment. No documentation could be found on the game and no one had any idea where the game originated. Also, it could not be determined if the game had adequate AI to make intelligent and more importantly consistent moves. Another reason this game was not used in the experiment was because it was found to have some quite harmful end-game logic flaws which produced poor computer moves.

corresponding to the master experimental schedule, and would then select an initial board set-up. Four game board set-ups were pre-programmed into the experiment. The umpire additionally had the option of entering an arbitrary set-up in case something had gone wrong and a game had to be resumed at some place other than the initial set-up.

b. Once the umpire had finished initializing the game, the player would then be asked to enter his or her name and experience level. There were four different experience levels the player could choose. [Figure IV-5]

c. A pre-game briefing was then conducted by the umpire. The player's terminal would display the initial game board set-up, and explain to the player how much intelligence would be provided during this game. The umpire would insure that the player fully understood what was being displayed and also inform the player of the time allocated to make all subsequent moves. Finally the player was told what pieces were already captured and advised to make the first move when familiar with the pieces' positions on the board.

5. Board Display of Intelligence Levels

Different combinations of the two fundamental board representations shown in Figures III-2 & III-3 were used to display the six choices of intelligence which could be provided to the player. The intelligence given to the player when playing the game at intelligence level six is shown in Figure IV-6-f. This representation shows the greatest amount of intelligence a player can receive. All other levels are made up of subsets of the level six display.

The board shown in Figure IV-6-a is a representation of level 1, the normal game board. It shows all of the white and black pieces. This is the only board displayed under level 1. Intelligence level 2 [Figure IV-6-b] displayed the same situation as level 1, but without any of the black pieces displayed. Level 3 [Figure IV-6-c] included the same display as shown in level 2, with the addition of a display which showed all of white's safe moves. The level 4 display [Figure IV-6-d] showed all of white's pieces and those black pieces that white was in a position to attack in a single move. Figure IV-6-e illustrates level 5. That display combines the attack board of level 4 with the safe board of level 3.

Figures IV-6-a through f are depictions of the situation presented to the player at the beginning of a game under board set-up three. Since intelligence levels 2 and 3 were not used for the actual experiment, there were four possible displays of each of the four board set-ups, for a total of 16 different views a subject might see when play began. [Figure IV-7]

6. Executing a Game Turn

A game turn consisted of one move each by the player and the umpire. The player's turn would begin when the player had received the last move by the umpire and the playing board(s) had been updated. The subject would then have a maximum of two minutes to review the information provided and make the next move. The only exception to this timing requirement was the first move. Before the first move the subject would have as much time as desired to study the initial board position and make the first move.

Once the player had made a move, the umpire would receive that move on the umpire terminal and enter the move into the Chess Challenger. The Chess Challenger would then derive a move for black. If no additional data collection was required during the turn, the umpire would enter Chess Challenger's move into the computer and the boards would be updated for the player's next move. If data collection was required for that move, the umpire would be prompted by the terminal to enter an e. luation code which described the subject's board strength at that particular time of the game. This code would be obtained from the Chess Challenger and entered into the computer. The board would then be updated and a new turn would begin.

7. Data Collection

The evaluation code was automatically collected and recorded at game turns eight, nine, and ten. At game turn 10 the entire board was recorded. Additionally, by the use of the SAVE DATA function built into the software, data could be captured at any point during the game, at the umpire's request.

E. ERROR CORRECTION AND RECOVERY

There was no "take-back" or "whoops" command built into the software to enable a player to retract back a legal move that was already entered. As in the real game of chess, once a move was entered it could not be changed. There were, however, ways to correct errors in entries if necessary. The procedure used most commonly when an error required correcting or the computer went down was to reset the board and set up the initial board positions to the situation of the board before the incorrect move.

V. EVALUATION OF DATA

A. COLLECTING THE DATA

As mentioned earlier, our software automatically created a data file for each move. On the first move the subject's name and experience level, the intelligence level and initial board setup being played against, and a representation of the board at that instant were recorded. On all moves, the moves of White and Black in standard chess alphanumeric format were saved along with the elapsed time from when White got the W. prompt until a legal move had been correctly entered. At the eigth, ninth, and tenth moves the software also queried the umpire for an evaluation score which was obtained from the Chess Challenger. The evaluation took the form of a six character alphanumeric representation unique to the Chess Challenger and a "B" or "W" to indicate advantage to Black or White. At the tenth move the software also recorded the subject's name and experience level, the intelligence level and initial board setup being played against, and a representation of the board at that instant.

The conversion of the evaluation code captured at moves eight, nine, and ten to our numerical measure of effectiveness (MOE) was a four step process.

Step 1. Each of the four initial board setups were put into the Chess Challenger to obtain a baseline evaluation for that setup. Using a table in the Operator's Manual for the Chess Challenger, the six character evaluation was decoded into a numerical score. A score showing advantages to White was recorded as positive; advantage to Black was negative.

Step 2. The data files were printed out and the six character evaluation code at the tenth move was similarly decoded into a raw score. Because all of our subjects were losing at move ten, all the raw scores were negative.

Step 3. The penalty for excessive time for the subject to enter moves was calculated. This was done by observing on the data file the elapsed time for White's moves on the second through tenth moves. Any times greater than 120 seconds per move were summed to obtain a total penalty time in seconds. For each minute, or fraction thereof, of penalty time the subject lost a number of points equal to the value of one pawn (256 points). For example, a total penalty time of 75 seconds, or 1.25 minutes, results in a penalty of 2 X 256 = 512 points.

Step 4. The raw score obtained in Step 2 was adjusted for time penalties and initial setup advantage by subtracting the results of Steps 1 and 3 to arrive at the MOE. Note that an initial setup advantage to Black, a negative number, causes the MOE to be more positive because a negative number is subtracted. This is as it should be because it rewards White for overcoming an initial disadvantage.

Three occasions arose where the subject was checkmated before move ten. An arbitrarily large negative score of -99999 was assigned in those cases and used as the MOE.

A new data file containing six columns was then built and is included as Table V-1. The first column is a subject identification number that matched each subject's name. Since the specific performance of any particular individual was not an issue, the corresponding names were not provided. Column 2 is the representation of each subject's chess playing experience: "1" for a complete novice, "2" for the subject who was familiar with chess but not a regular or frequent player, "3" for a frequent player, and "4" for a tournament player. The data in column 3 is the intelligence level presented to the subject on that trial. The data in column 4 gives the initial board setup for each trial. Column 5 contains the MOE. The value in column 6 indicates the sequence in which this trial occurred, i.e., whether it was this subject's first, second, third, or fourth trial.

B. ANALYSIS

1. Initial Quick Look

The first look at the MOE data showed huge variances that resulted from six outliers in the one hundred twenty-four trials. Of those six, three were cases in which the subject was checkmated. The other three were instances where checkmate was imminent. The largest of these scores was -30556; the smallest of the remaining scores was -7151. The six exaggerated scores were made by five different subjects, against three of the four intelligence levels and three of the four initial setups, and occurred on the second through fourth trials. In other words, they appear to be randomly dispersed.

2. Handling The Dilemma

Proper treatment of these outliers was necessary to proceed further with any statistical analysis. After investigating several potential paths we decided to recode the six exaggerated scores to a value lower than the lowest in the main body of data points but not so disastrously low as that initially coded. We selected, arbitrarily, the value -9000.

We feel this was a reasonable approach because the MOE for a checkmate was arbitrarily set and the value we picked was sufficiently large to set it off from those MOE's arrived at otherwise. This method allowed continued analysis without reducing the size of our data base while preserving the significantly more disastrous results on those six trials.

All further analysis and conclusions refer to this "adjusted data."

3. Determine Which Parameters Were Significant

The next step was to determine which of the factors in the mathematical model were statistically significant. To do this we used a general linear model procedure known as the "Extra Sum of Squares" method [Ref. 9].

The basic idea of this method is to do an analysis of variance (ANOVA) using the entire model. Then repeat the ANOVA on a reduced model that omits the parameters corresponding to the factors under investigation. The difference in the model sum of squares for the two runs is due to the influence of this factor. Using the two sums of squares an F statistic is then calculated and used to indicate the significance of the factor or factors under consideration. The equation is:

$$F = \frac{[RSS(f) - RSS(?)] \,/\, [DF(f) - DF(?)]}{ESS(f) \,/\, [DF(t) - DF(f)]}$$

where:

RSS(f) = Regression Sum of Squares, full model

RSS(?) = Regression Sum of Squares, modified model

DF(f) = Degrees of Freedom of regression, full model

DF(?) = Degree of Freedom of regression, modified model

DF(t) = Total Degree of Freedom, same for either model

ESS(f) = Error Sum of Squares, full model

The specific values and results of the computations appear in Table V-2. At a 95% confidence level, the intelligence level, and the initial board setup were both statistically significant. The low calculated F-statistic for trial number shows that the experiment design successfully precluded "learning" from effecting the results. Also as expected, the subject's experience as a chess player was significant. Inspection shows that those with the most experience scored highest.

The intent in the experimental design had been to make the initial board setup insignificant. Since the results showed this was not the case, board setup was investigated further as was the effect of intelligence level provided.

4. <u>Critical Factors in the Significant Parameters</u>

To determine which levels were significantly different in the factors intelligence level and board setup we used the Scheffe multiple comparison analysis of variance procedure.

The basic idea of "Scheffe's Test" is to compare the means of the samples of concern two at a time in all combinations of two and arrive at simultaneous 95% confidence levels for the differences of any pair of levels of a factor. As an example, a Scheffe's multiple comparison of three sample populations would say that with 95% confidence, all the following statements are true:

$$(\mu_1 - \mu_2) = (\overline{X}_1 - \overline{X}_2) \pm \sqrt{(r-1)\ F\ .05}\ \ S_p \sqrt{\frac{1}{n_1} + \frac{1}{n_2}}$$

$$(\mu_1 - \mu_3) = (\overline{X}_1 - \overline{X}_3) \pm \sqrt{(r-1)\ F\ .05}\ \ S_p \sqrt{\frac{1}{n_1} \quad \frac{1}{n_3}}$$

$$(\mu_2 - \mu_3) = (\overline{X}_2 - \overline{X}_3) \pm \sqrt{(r-1)\ F\ .05}\ \ S_p \sqrt{\frac{1}{n_2} \quad \frac{1}{n_3}}$$

where

$F_{.05}$ = critical value of F (with r-1 and r(n-1) degress of freedom) leaving 5% in the upper tail

S_p^2 = pooled standard deviation, $\frac{1}{r} \sum_{i=1}^{r} S_i^2$

r = number of means to be compared

n_i = sample sizes.

Wonnacott and Wonnacott [Ref. 10] provides a good illustration.

The results of Scheffe's Test for Intelligence level showed that levels 1 and 3 were significantly different from each other but neither varied significantly from levels 5 and 6. When applied to the initial board setup, scores against setup 3 were significantly worse than against setups 1, 2, or 4. Table V-3 provides the data leading to these conclusions.

C. WHAT WENT WRONG WITH SETUP-3?

We knew from our evaluation of the initial board setups that, as far as the Chess Challenger was concerned, setup 3 was the second most advantageous for White so the answer was not in the numerical realm. We had been the umpire for approximately 95% of the trials and began to think about what we had observed while the subjects faced that setup. In comparing notes we found that we both had observed many instances of our subjects falling into an unintentional trap in the first few moves. From the initial positions shown in Figure II-3, White almost always made the apparently optimal move of queen takes rook at QB3 to which Black responded by knight takes pawn at White's K4. Probably due to their lack of chess skills and the unfamiliarity of our symbolic board representation, the vast majority of our subjects failed to recognize that Black's knight now forked their queen and the bishop at KN5. Many saw the threat to the bishop only or the threat to their knight at KB3 from Black's bishop and moved accordingly. Black then captured White's queen and the victim never recovered from the sudden early loss. On several occasions it seemed the psychological impact of the queen's loss at this stage was so staggering to the subject that it was worse than the material loss. Perhaps a similar thing happened to the numerically superior French army in 1939 when the German army swept around the Maginot Line. After the Germans rendered useless what had been the centerpiece of the French defense, the French army was quickly defeated.

D. WHAT IF SETUP-3 IS OMITTED?

The Extra Sum of Squares and Scheffe's Test procedures were repeated on a modified data file that omitted all trials against initial setup 3. Again, the intelligence levels 1 and 3 were different from each other. Neither was statistically different from levels 5 or 6. Initial board setup was not significant. Our interpretation of these results is discussed with the rest of our conclusions in Chapter VI.

VI. CONCLUSIONS AND RECOMMENDATIONS

A. CONCLUSIONS

1. An Optimum Amount of Information Exists

Performance on the simulated battlefield tends to improve as the overall amount of information about one's opponent increases up to some optimum level. We observed that as our subjects were given more information about Black's strength and position their scores improved until, at some point, the additional information was too much to be effectively utilized in the time allowed. Scores against Intelligence Level 3 (the least information) were significantly worse than when the next higher amount of information was presented in Intelligence Level 5. Scores against Intelligence Level 5 were similarly not as high as against the next higher amount of information provided in Intelligence Level 1 (the normal view of the board). Additional information beyond that point served only to confuse the situation. This resulted in degraded performance. Scores against Intelligence Level 6, which displayed the most information, were significantly lower than against Intelligence Level 1. The possible reasons for this are multiple.

On the one hand, the additional information may be simply too much information to be assimilated in the time allotted. A direct analogy can be drawn to a military command center into which messages flow at a faster rate than they can be digested or acted upon. They pile up all over the command center perhaps obscuring other information. Important data gets lost with the general deluge because it cannot be spotted and separated from the chaff.

Another possibility is that the total amount of information is not necessarily excessive but that in the format in which presented it is excessive. To illustrate this point consider the information contained in this paragraph. It can be easily read and understood in a few moments. However, if the same amount of information (i.e., *this paragraph*) were given to the reader as a block of dots and dashes along with a copy of the International Morse Code the average reader would have significant difficulty understanding it. It is important to note that our experiment was not about the method of presentation but the quantity of information presented. Within that context, our results still hold. There will be some optimum amount of information that can be utilized by a particular subject for each separate method of information presentation. Beyond that point too much time is spent in deciphering the presentation to allow adequate time for digesting it and formulating a plan of action.

In the limit, of course, there will exist some quantity of data that is excessive regardless of the method of presentation. We have the physical ability to pass that saturation threshold now by stacking teletype machines and communications systems in our command centers. We also have a tendency to overkill at every level. No Captain wants to tell the Admiral, "I don't know", when asked a question so the Captain ensures the information is there to cover any area about which the Captain thinks the Admiral might ask. Likewise, the Lieutenants to whom the Captains turn with their questions try to ensure they will always have the answers available. And so the quantity of information we may think we desire continues to mushroom. A very real and continuing problem of modern warfare is how to adequately balance the capability to provide information, desirability of having given data, and the optimum display of the information that is desired. The interaction between the method of display and the amount of information becomes increasingly important as the amount of information desired becomes larger.

2. Experience and Training Help

We observed the subjects with more chess playing experience tended to score higher than subjects with less experience. A direct analogy can be drawn to the battlefield. To exaggerate the obvious, one would not expect a new second lieutenant to fare as well directing an army as an infantry lieutenant general with thirty years' experience. Likewise, one would expect a vice admiral of similar experience to fare better in command of a carrier battle group than the lieutenant general would.

3. Psychological Impact Can be a Major Factor

As mentioned earlier, on several occasions while playing against setup 3 our subjects lost their Queen in the first few moves. This was a significant material loss in each instance. But in some cases the psychological impact seemed even more devastating. Some subjects were visibly upset for several moves afterward and never regained control of the situation. They were thereafter unable to mount a coordinated attack. A direct analogy can be drawn to the effect of a serious loss early in an engagement. As a hypothetical situation, consider a carrier battle group preparing for an approaching air raid that should be easily repulsed. Just as the incoming bombers are detected and the anti-aircraft plan starts to unfold the carrier suffers an internal explosion from dropped ordinance and is put out of action. Despite the loss of the carrier, the remaining fighter aircraft and surface combatants should be able to repulse the raid. The remaining escorts could then nurse the carrier back to a safe area and regroup for further action. However, it is conceivable that the early critical damage to the carrier could cause significant disorientation of the defense manifested in screen disintegration and in wasted time and effort to find and combat a nonexistent submarine threat (this would not have to be prolonged but simply a distraction from the task at hand). The result could be significantly greater effectiveness of the air raid.

B. CONCLUSIONS THAT CAN NOT BE REACHED

Why was our optimal amount of information optimal? Was it because the absolute amount of information given to the subjects and the method of display were in proper balance or was it because that presentation most nearly resembled the normal view of a chess board with which our subjects were all somewhat familiar? We cannot answer that question from our experiment. We suspect that familiarity was a factor in making that particular display optimal. However, it can be argued that the amount of information was still the major factor because the same display was included as a portion of the information level 6 display against which our subjects scored more poorly.

C. RECOMMENATIONS FOR FUTURE WORK

1. Expand the Sample Size

With a larger sample space one would expect the results to become more clear cut. Perhaps the adjustments we had to perform on the outlying scores could be done away with and those points omitted. With the small original sample size, omission of those scores produced inconclusive results. The new samples could also be analyzed as a separate group and those results compared to the original.

2. <u>Compare Methods of Information Display</u>

The experiment could be run using only one level of information but displaying it in a variety of ways. The display methods could include:

a) the same as in this experiment

b) the same as in this experiment but allow the subject to use a standard chess board and pieces for manual manipulation as a decision or visualization aid

c) use the RISNEY/TSCHUDY project from OS-4602, C3 Systems Evaluation (Fall Qtr 1983), to display the chess board and pieces as iconic symbols on the RAMTEK color monitors in the C3 laboratory. This software produces an easily manipulated computer generated color graphic representation of the board with standard shapes for all the pieces.

3. <u>Start From the Opening Move of the Game</u>

This experiment started the subject at mid-game with the explicit intention of denying the subject any prior intelligence as to the exact strength and disposition of the opponent. The experiment could be run with the game always beginning at the first move and proceeding for some longer number of moves well into middle-game. The number of moves to play would have to be determined by trial runs to re-establish a good sample point. A potentially confounding element that must be investigated is the impact of chess playing experience. The better players would be expected to play a better opening game. This could drastically affect the number of moves before checkmate and therefore the appropriate sample point. A large enough sample set of experienced players with nearly equal ratings may be able to avoid the problem.

4. Test the Relation Between Experience and the Amount or Method of Display of Information

Based on subjective observation by the umpires, when playing against information level 6 the more experienced players relied less on the safe position and possible attack portions of the display while inexperienced players used them heavily. That hypothesis could be tested but the difficulty would lie in how to measure utilization of the various portions of data displayed. With the equipment currently available at the Naval Postgraduate School, that could only be done very subjectively with questionnaires for the subjects. Though not available here, there exist in commercial use devices for accurately measuring how the human eye scans an area. These could be used to quantitatively examine the percentage of time a subject actually looked at any given sector of the display.

Other related experiments are certainly possible. Our experiment was never contemplated as exhaustive, but more as a beginning. The field of information management is becoming ever more complicated and ever more important. Therefore, the potential for experiments such as this to serve a useful purpose increases. Perhaps the same idea could be used in specific applications to improve the utility of information displays in command centers if adequate control of experimental factors could be established.

APPENDIX A

FIGURES

```
-- BR BB BQ BR ** BK **
** BP BP -- ** BP BB BP
BP ** BN BP BP BN BP **
** -- ** -- ** -- ** --
-- ** WP WP WP ** -- **
** WP WN -- WB WP ** --
WP ** -- WQ WN ** WP WP
** -- WR -- WK WB ** WR
```

Figure II-1
INITIAL SCENARIO NO.1

```
BR ** -- ** -- BR BK **
BP BP ** BN BB BP BP --
-- BQ -- ** BB BN -- BP
** -- BP BP ** -- ** --
-- ** -- WP -- ** -- WB
** -- WP WB ** WN ** WP
WP WP -- WN -- WP WP **
WR -- ** WQ WR -- WK --
```

Figure II-2
INITIAL SCENARIO NO.2

```
-- ** -- BQ -- BR BK **
** -- BP -- BN BP BB BP
-- ** -- BP -- BN BP **
** BP ** WP BP -- WB --
-- WP -- ** WP ** BB **
** -- BR -- ** WN ** --
-- ** WP WQ WB WP WP WP
** -- ** WR ** WR WK --
```

Figure II-3
INITIAL SCENARIO NO.3

```
BR ** BB BK -- BB -- BR
BP -- ** -- ** BP BP BP
-- ** BP ** -- BN -- **
** WN ** -- ** -- ** --
-- ** -- ** -- ** -- **
** -- WN -- ** -- ** --
WP WP -- ** WP WP WP WP
WR -- ** -- WK WB ** WR
```

Figure II-4
INITIAL SCENARIO NO.4

111	112	113	114	115	116	117	118	119	120
101	102	103	104	105	106	107	108	109	110
91	92	93	94	95	96	97	98	99	120
81	82	83	84	85	86	87	88	89	90
71	72	73	74	75	76	77	78	79	80
61	62	63	64	65	66	67	68	69	70
51	52	53	54	55	56	57	58	59	60
41	42	43	44	45	46	47	48	49	50
31	32	33	34	35	36	37	38	39	40
21	22	23	24	25	26	27	28	29	30
11	12	13	14	15	16	17	18	19	20
1	2	3	4	5	6	7	8	9	10

Figure III-1.
120 Element Array Representation of the game board

99	99	99	99	99	99	99	99	99	99
99	99	99	99	99	99	99	99	99	99
99	-4	-2	-3	-5	-6	-3	-2	-4	99
99	-1	-1	-1	-1	-1	-1	-1	-1	99
99	0	0	0	0	0	0	0	0	99
99	0	0	0	0	0	0	0	0	99
99	0	0	0	0	0	0	0	0	99
99	0	0	0	0	0	0	0	0	99
99	1	1	1	1	1	1	1	1	99
99	4	2	3	5	6	3	2	4	99
99	99	99	99	99	99	99	99	99	99
99	99	99	99	99	99	99	99	99	99

Figure III-2.
Internal Board Representation
of the Default Initial Game Positions

```
BR BN BB BQ BK BB BN BR
BP BP BP BP BP BP BP BP
** -- ** -- ** -- ** --
-- ** -- ** -- ** -- **
** -- ** -- ** -- ** --
-- ** -- ** -- ** -- **
WP WP WP WP WP WP WP WP
WR WN WB WQ WK WB WN WR
```

Figure III-3.
Normal Game Board as Displayed on the Terminal

```
QR8                 K8
QR7                          KN7
QR6                          KN6 KR6
        QB5         K5       KN5 KR5
QR4 QN4 QB4         K4       KN4 KR4
QR3 QN3         Q3  K3       KN3 KR3
QR2     QB2     Q2  K2       KN2 KR2
QR1 QN1 QB1     Q1  K1  KB1  KN1 KR1
```

Figure III-4.
Example of the Safe Board Display

```
Repeat until an Exit from program is Requested.
    Introductory Section.

    Repeat until a Reset or Exit is Requested.
        Determine Intelligence.
        Display Intelligence to User.

        Repeat until not Illegal Move.

            Repeat until not Illegal Move.
                Read in Move.
                Parse Move.
            End Repeat.

            If Regular Move, Determine if Legal;
                Else if Castle, Determine if Legal.

        End Repeat.

        Output Data Collection Information.
        Reverse Turn.
    End Repeat.

End Repeat.
```

Figure III-5.
Control Flow

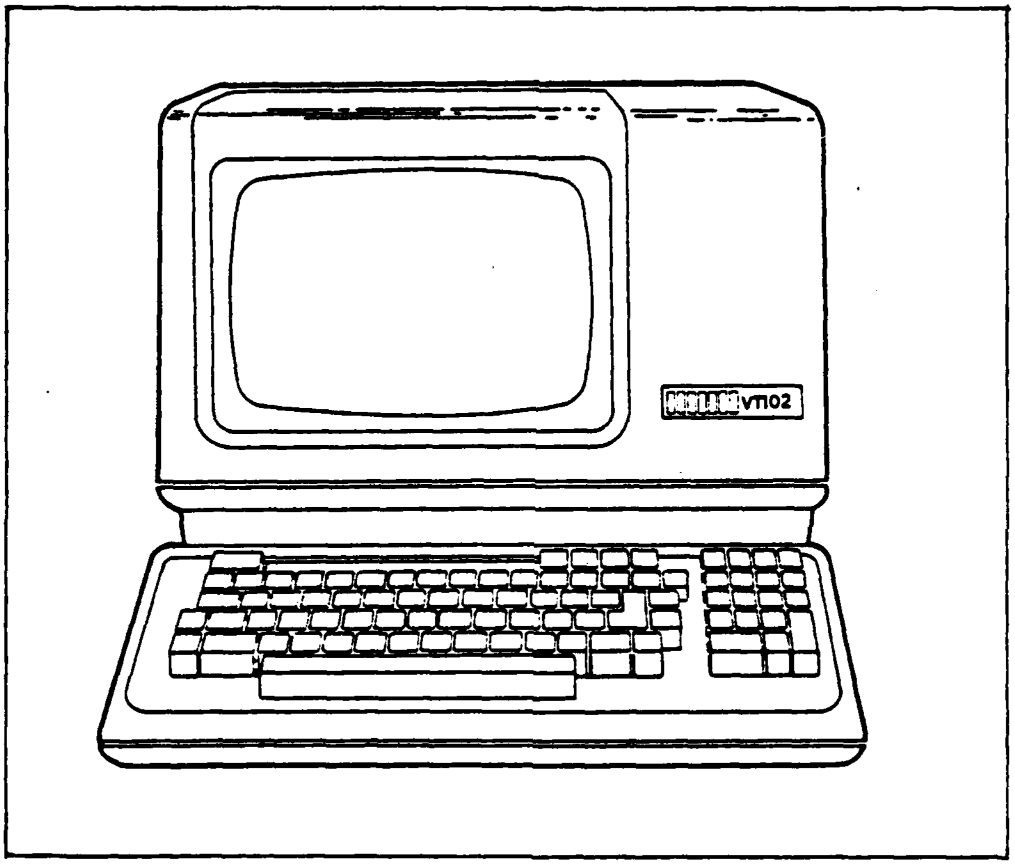


Figure IV-1.
VT 100/102 Video Display Terminal

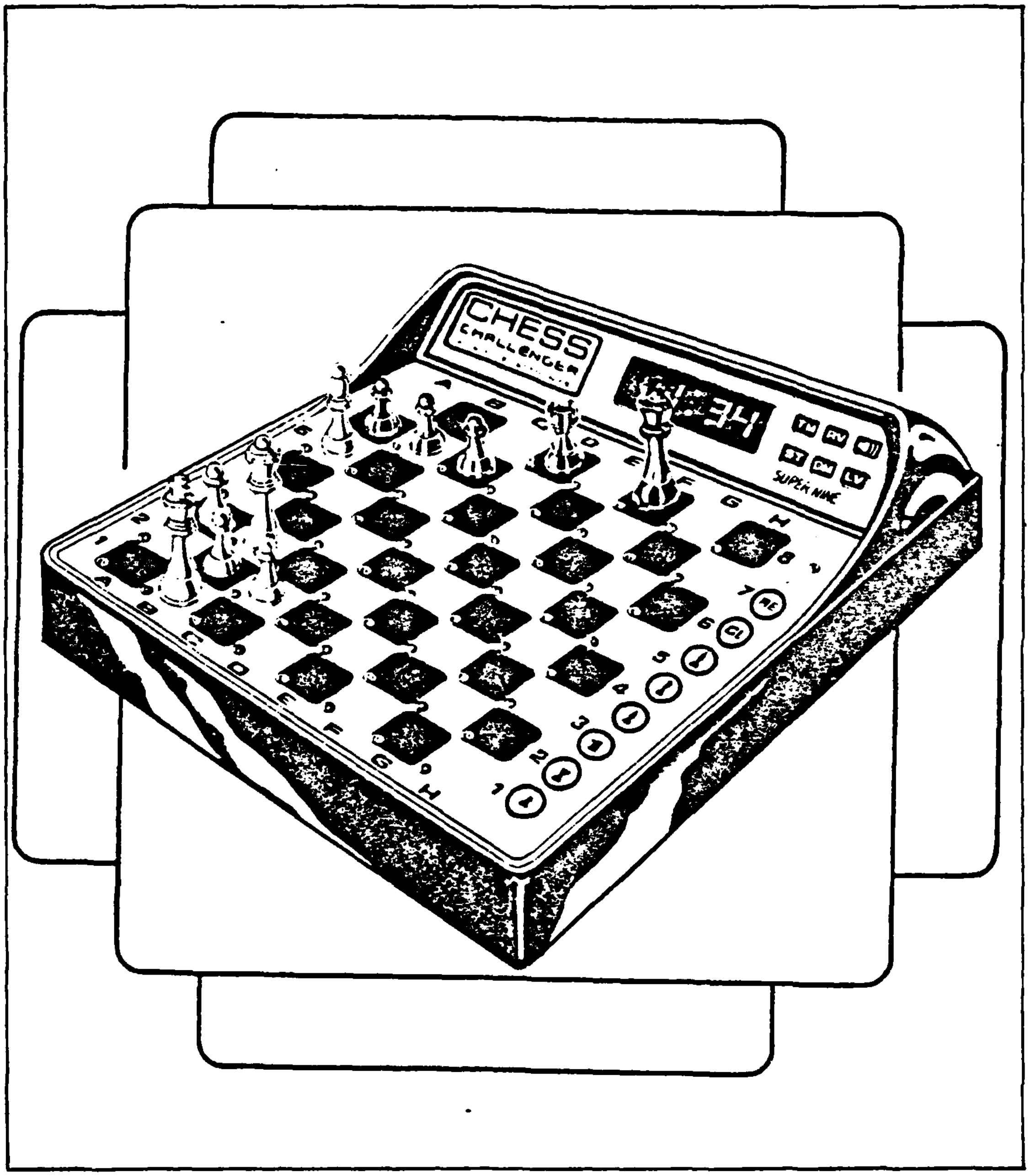


Figure IV-2.
Fidelity Electronics Super Nine Chess Challenger

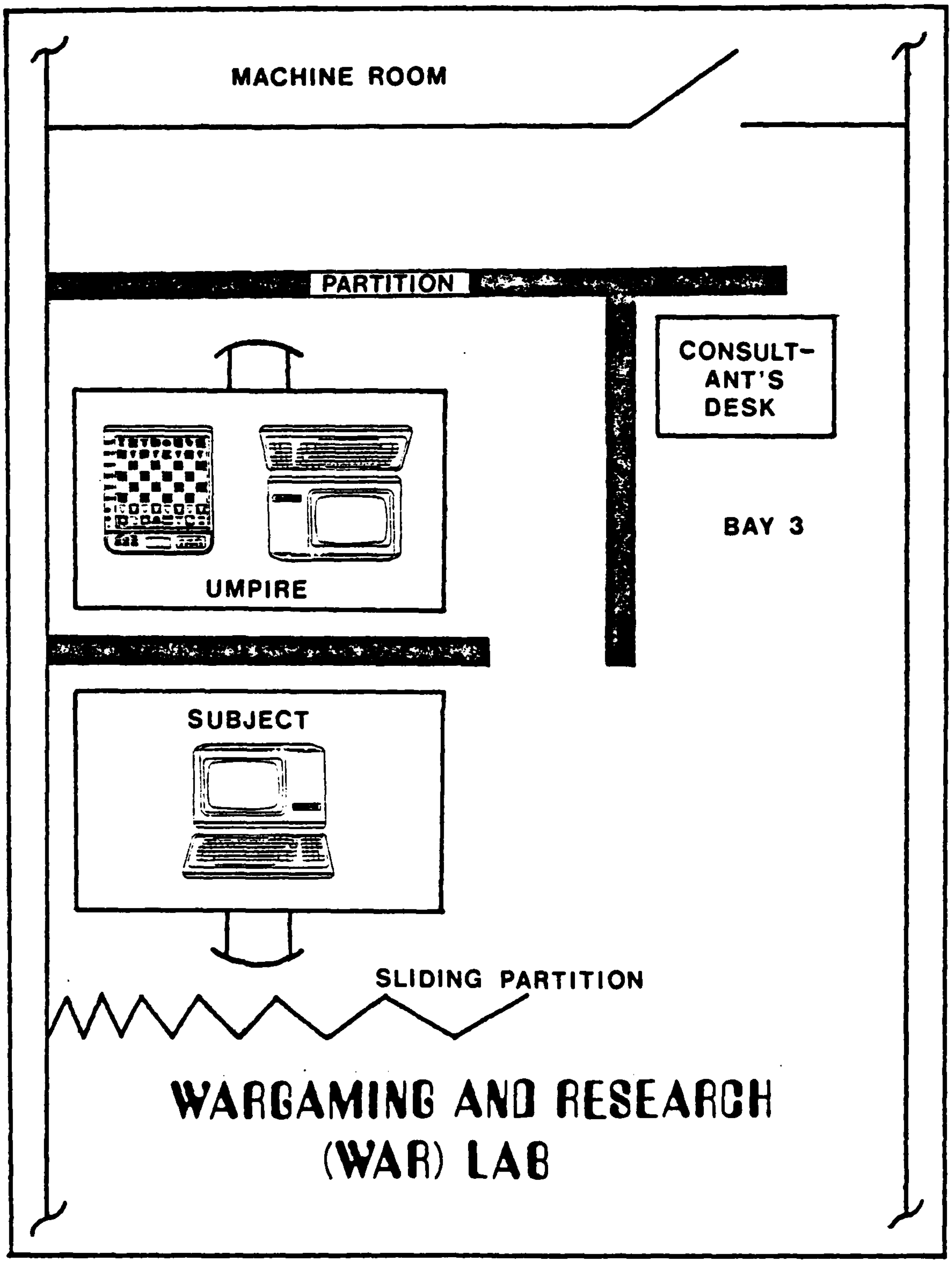


Figure IV-3.
Experiment Layout

```
HELLO AND WELCOME TO CHESS

      TO INITIALIZE THIS PROGRAM SELECT THE TYPE
      OF BOARD DISPLAY FOR THE PLAYER.

        1.-- DISPLAY ENTIRE BOARD

        2.-- DISPLAY JUST WHITE PIECES

        3.-- DISPLAY WHITE PIECES AND BLACK'S PIECES
             THAT CAN BE ATTACKED

        4.-- DISPLAY WHITE PIECES AND WHITE SAFE
             MOVES

        5.-- DISPLAY WHITE PIECES, WHITE SAFE MOVES,
             AND BLACK PIECES THAT CAN BE ATTACKED

        6.-- DISPLAY ENTIRE BOARD, BLACK PIECES THAT
             CAN BE ATTACKED, AND WHITE SAFE MOVES
```

Figure IV-4.
Menu Selection for Different Intelligence Levels

```
NOW PLEASE ENTER YOUR EXPERIENCE LEVEL

   A.-- NEVER HAVE PLAYED CHESS BEFORE
   B.-- A NOVICE CHESS PLAYER
   C.-- PLAY CHESS FREQUENTLY
   D.-- TOURNAMENT CHESS PLAYER
```

Figure IV-5.
Experience Level Menu

```
+------------------------------------+
|                                    |
|          NORMAL BOARD              |
|                                    |
|    -- ** -- BQ -- BR BK **         |
|    ** -- BP -- BN BP BB BP         |
|    -- ** -- BP -- BN BP **         |
|    ** BP ** WP BP -- WB --         |
|    -- WP -- ** WP ** BB **         |
|    ** -- BR -- ** WN ** --         |
|    -- ** WP WQ WB WP WP WP         |
|    ** -- ** WR ** WR WK --         |
|                                    |
+------------------------------------+
```

Figure IV-6-a.
Level 1: Normal View of the Board

```
+------------------------------------+
|                                    |
|          WHITE PIECES              |
|                                    |
|    -- ** -- ** -- ** -- **         |
|    ** -- ** -- ** -- ** --         |
|    -- ** -- ** -- ** -- **         |
|    ** -- ** WP ** -- WB --         |
|    -- WP -- ** WP ** -- **         |
|    ** -- ** -- ** WN ** --         |
|    -- ** WP WQ WB WP WP WP         |
|    ** -- ** WR ** WR WK --         |
|                                    |
+------------------------------------+
```

Figure IV-6-b.
Level 2: White Pieces Only

```
WHITES AND ATTACKS

-- ** -- ** -- ** -- **
** -- ** -- ** -- ** --
-- ** -- ** -- BN -- **
** BP ** WP BP -- WB --
-- WP -- ** WP ** -- **
** -- BR -- ** WN ** --
-- ** WP WQ WB WP WP WP
** -- ** WR ** WR WK --
```

Figure IV-6-c.
Level 3: White Pieces and Black Pieces that White can Attack

```
WHITE PIECES                          SAFES
                           ------------------------------------------
-- ** -- ** -- ** -- **
** -- ** -- ** -- ** --    QR7  QN7
-- ** -- ** -- ** -- **    QR6
** -- ** WP ** -- WB --    QR5  QN5                          KN5
-- WP -- ** WP ** -- **         QN4                                KR4
** -- ** -- ** WN ** --              QB3                     KN3
-- ** WP WQ WB WP WP WP    QR2  QN2        Q2   K2   KB2     KN2   KR2
** -- ** WR ** WR WK --    QR1  QN1  QB1   Q1   K1   KB1     KN1   KR1
```

Figure IV-6-d.
Level 4: White Pieces and White Safe Moves

```
+----------------------------------------------------------------------------------+
|                                                                                  |
|   WHITES AND ATTACKS                                 SAFES                       |
|                                      ---------------------------------------    |
|  -- ** -- ** -- ** -- **                                                         |
|  ** -- ** -- ** -- ** --            QR7  QN7                                     |
|  -- ** -- ** -- BN -- **            QR6                                          |
|  ** BP ** WP BP -- WB --            QR5  QN5                      KN5            |
|  -- WP -- ** WP ** -- **                 QN4                           KR4       |
|  ** -- BR -- ** WN ** --                      QB3                 KN3            |
|  -- ** WP WQ WB WP WP WP            QR2  QN2       Q2   K2   KB2  KN2  KR2       |
|  ** -- ** WR ** WR WK --            QR1  QN1  QB1  Q1   K1   KB1  KN1  KR1       |
|                                                                                  |
+----------------------------------------------------------------------------------+
```

Figure IV-6-e.
Level 5: White Pieces, Black Pieces that White can Attack, and White Safe Moves

```
NORMAL BOARD

-- ** -- BQ -- BR BK **
** -- BP -- BN BP BB BP
-- ** -- BP -- BN BP **
** BP ** WP BP -- WB --
-- WP -- ** WP ** BB **
** -- BR -- ** WN ** --
-- ** WP WQ WB WP WP WP
** -- ** WR ** WR WK --
```

```
                         SAFES
---------------------------------------------------------
QR7  QN7
QR6
QR5  QN5                                KN5
     QN4                                       KR4
          QB3                           KN3
QR2  QN2        Q2     K2     KB2       KN2    KR2
QR1  QN1  QB1   Q1     K1     KB1       KN1    KR1
```

```
ATTACKS

-- ** -- ** -- ** -- **
** -- ** -- ** -- ** --
-- ** -- ** -- BN -- **
** BP ** WP BP -- WB --
-- WP -- ** WP ** -- **
** -- BR -- ** WN ** --
-- ** WP WQ WB WP WP WP
** -- ** WR ** WR WK --
```

Figure IV-6-f.
Level 6: Normal Board, Black Pieces that can be Attacked by White, and White Safe Moves

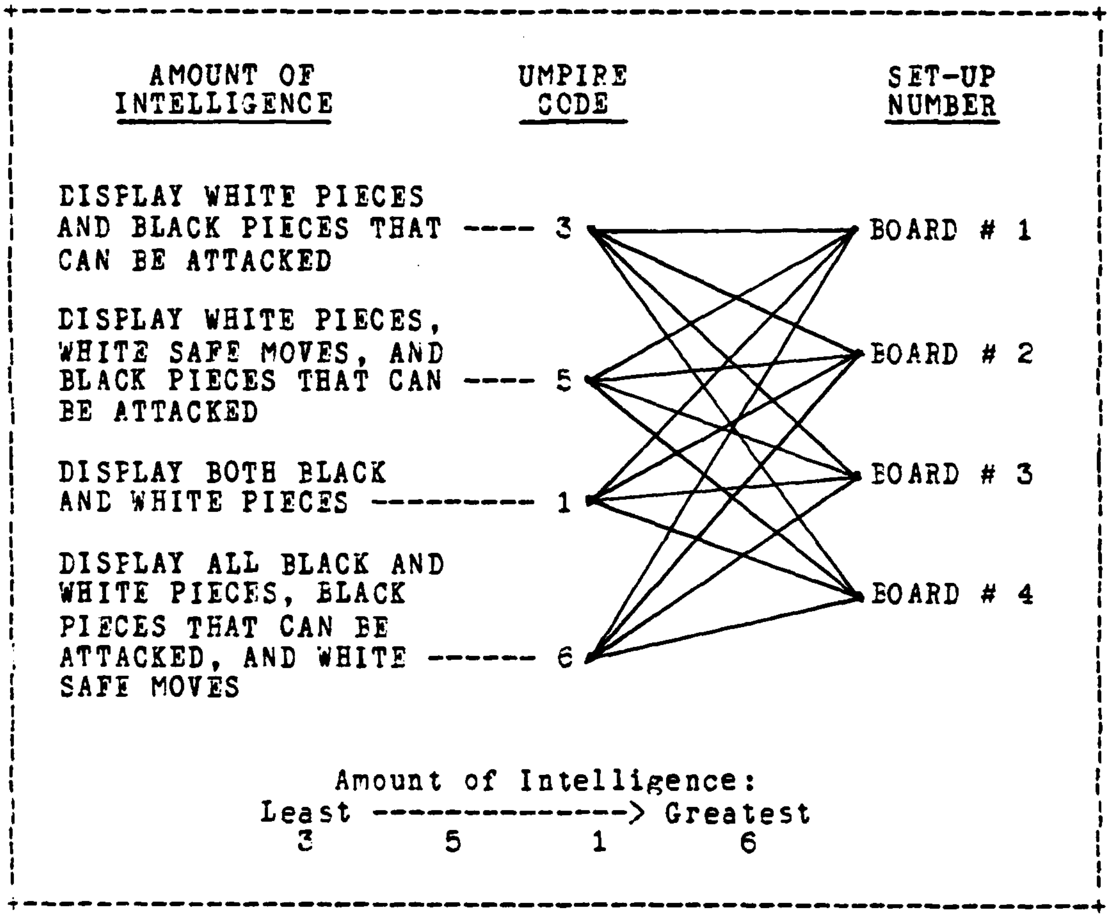


Figure IV-7.
The 16 Possible Game Combinations.

APPENDIX B

TABLES

TABLE III-1

PLAYING PIECE REPRESENTATIONS

NUMBER	COLOR	REPRESENTATION
Ø		Empty
+1	White	Pawn
+2	" "	Knight
+3	" "	Bishop
+4	" "	Rook
+5	" "	Queen
+6	" "	King
-1	Black	Pawn
-2	" "	Knight
-3	" "	Bishop
-4	" "	Rook
-5	" "	Queen
-6	" "	King
99		Off the Board

TABLE V-1

EXPERIMENT DATA FILE

SUBJECT ID		EXP LVL	INT LVL	SET UP	SCORE	TRIAL
1	________	2	1	1	-3237	1
2	________	1	5	1	-1155	1
3	________	2	3	1	-1477	1
4	________	2	6	1	-1834	1
5	________	2	1	1	-302	1
6	________	2	5	1	-319	1
7	________	2	3	2	-2181	1
8	________	2	6	2	-2317	1
9	________	3	1	2	-2041	1
10	________	2	5	2	-5722	1
11	________	2	3	2	-1901	1
12	________	2	6	2	-2915	1
13	________	2	1	3	-5558	1
14	________	2	5	3	-5968	1
15	________	2	3	3	-2896	1
16	________	2	6	3	-6812	1
17	________	2	1	3	-3351	1
18	________	1	5	3	-6207	1
19	________	2	3	4	-2599	1
20	________	3	6	4	-2308	1
21	________	2	1	4	-1481	1
22	________	2	5	4	-3058	1
23	________	2	3	4	-1000	1
24	________	2	6	4	-2770	1
25	________	2	1	1	-682	1
26	________	2	5	1	-2319	1
27	________	1	3	1	-1907	1
28	________	2	6	1	-2907	1
29	________	2	1	1	-1408	1
30	________	3	5	1	-2643	1
31	________	2	3	2	-4320	1
1	________	2	5	2	-3963	2
2	________	1	1	2	-3480	2
3	________	2	1	3	-3337	2
4	________	2	1	3	-2711	2
5	________	2	5	4	-2473	2
6	________	2	1	4	-3026	2
7	________	2	1	1	-1263	2
8	________	2	1	1	-1618	2
9	________	3	3	3	-1545	2
10	________	2	3	3	-3731	2

TABLE V-1 [continued]

SUBJECT ID	EXP LVL	INT LVL	SET UP	SCORE	TRIAL
11	2	5	4	-1309	2
12	2	5	4	-2303	2
13	2	3	1	-2573	2
14	2	3	1	-33297	2
15	2	5	2	-1494	2
16	2	5	2	-593	2
17	2	6	4	-936	2
18	1	6	4	-2426	2
19	2	6	1	-3340	2
20	3	3	1	-1731	2
21	2	6	2	-1717	2
22	2	6	2	-1224	2
23	2	6	3	-4740	2
24	2	3	3	-3041	2
25	2	5	2	-2360	2
26	2	1	2	-4076	2
27	1	1	3	-2861	2
28	2	1	3	-859	2
29	2	5	4	-2219	2
30	3	1	4	-1033	2
31	2	1	1	-1887	2
1	2	3	3	-6247	3
2	1	3	4	-33135	3
3	2	5	2	-335	3
4	2	5	4	-1525	3
5	2	6	2	-3091	3
6	2	6	3	-2525	3
7	2	6	3	-1843	3
8	2	3	4	-3093	3
9	3	5	1	-334	3
10	2	1	4	-888	3
11	2	1	1	-2698	3
12	2	1	3	-2382	3
13	2	6	2	-1472	3
14	2	6	4	-1000	3
15	2	6	1	-489	3
16	2	3	4	-1904	3
17	2	5	1	-1947	3
18	1	1	2	-2995	3
19	2	1	2	-1654	3
20	3	1	3	-3844	3
21	2	3	1	-7151	3

TABLE V-1 [continued]

SUBJECT ID	EXP LVL	INT LVL	SET UP	SCORE	TRIAL
22	2	3	3	-99999	3
23	2	5	1	-1494	3
24	2	5	2	-1244	3
25	2	3	3	-1572	3
26	2	3	4	-2486	3
27	1	5	2	-2508	3
28	2	5	4	-2776	3
29	2	6	2	-2429	3
30	3	6	3	-4494	3
31	2	6	3	-3754	3
1	2	6	4	-1950	4
2	1	6	3	-99999	4
3	2	6	4	-2395	4
4	2	3	2	-3808	4
5	2	3	3	-5058	4
6	2	3	2	-5463	4
7	2	5	4	-99999	4
8	2	5	3	-30556	4
9	3	6	4	-1014	4
10	2	6	1	-3339	4
11	2	6	3	-947	4
12	2	3	1	-2897	4
13	2	5	4	-1006	4
14	2	1	2	-1621	4
15	2	1	4	-989	4
16	2	1	1	-2306	4
17	2	3	2	-3052	4
18	1	3	1	-652	4
19	2	5	3	-5307	4
20	3	5	2	-1356	4
21	2	5	3	-881	4
22	2	1	1	-1731	4
23	2	1	2	-1694	4
24	2	1	1	-1413	4
25	2	6	4	-1201	4
26	2	6	3	-3988	4
27	1	6	4	-1660	4
28	2	3	2	-5882	4
29	2	3	3	-2717	4
30	3	3	2	-1371	4
31	2	5	4	-1586	4

TABLE V-2

EXTRA SUM OF SQUARES DATA

RSS(f) = 114669799
ESS(f) = 387368583
DF(f) = 8
DF(t) = 115

Test for effect of:	RSS(?)	DF(?)	F(calc.)	Significance Level
INTELLIGENCE	86943082	5	2.74	0.0466
SETUP	48858404	5	6.51	0.0004
EXPERIENCE	97773847	7	5.02	0.0270
TRIAL NO.	114402961	7	0.08	0.7778

Note 1: Tabulated F-statistics are for the 95% confidence level.
Note 2: Those factors for which F(calculated) is greater than F(tabulated) are significant at the 95% confidence level.

TABLE V-3

SCHEFFE'S TEST RESULTS

Confidence Level = 95%

BY INTELLIGENCE LEVEL:

Grouping	Mean	INTELLIGENCE LEVEL
A	-2207.3	1 (normal board view)
B A	-2672.2	6 (most information)
B A	-2722.7	5
B	-3588.9	3 (least information)

Minimum significant difference = 1335.16

BY INITIAL BOARD SET UP:

Grouping	Mean	Set Up
A	-2195.3	1
A	-2335.9	4
A	-2589.6	2
B	-4070.2	3

Minimum significant difference = 1335.16

Note: The letters A and B in the Grouping columns have no special meaning. They serve only to illustrate which sample means are within simultaneous 95% confidence intervals of each other.

LIST OF REFERENCES

1. Joint Chiefs of Staff, Joint War Gaming Agency, Joint War Gaming Manual, JWGA-167-69, pp. 49-51, July 1969.

2. Digital Equipment Corporation, VAX-11 FORTRAN Language Reference Manual, April 1982.

3. Digital Equipment Corporation, Systems Services Reference Manual, May 1982.

4. Adelson-Velskey, G. M., Arlazarov, V. L., Bitman, A. R., Zhrvolovskiy, A. A., and Uskov, A. V., "Programming a Computer to Play Chess", Russian Math Surveys, v. 25, pp. 221-262, 1970.

5. Shannon, C. E., "Programming a Computer to Play Chess", Scientific American, v. 2, pp. 48-51, February 1950.

6. Frey, P. W., Chess Skill in Man and Machine, pp. 55-58, Springer-Verlag, 1977.

7. Levy, D., Chess and Computers, pp. 25-29, Computer Science Press, 1976.

8. Harkness, K., The Official Blue Book and Encyclopedia of Chess, p. 344, David McKay Company, Inc., 1956.

9. Draper, N. R., and Smith, H., Applied Regression Analysis, 2d ed., pp. 97-98, Wiley, 1981.

10. Wonnacott, Thomas H., and Wonnacott, R. J., Introductory Statistics, 3d ed., pp. 291-305, Wiley, 1977.

BIBLIOGRAPHY

Adelson-Velskey, G. M., Arlazarov, V. L., Bitman, A. R., Zhrvolovskiy, A. A., and Uskov, A. V., "Programming a Computer to Play Chess", Russian Math Surveys, v. 25, 1970.

Bardon, L., Play Better Chess, Octopus Books Limited, 1980.

Barr, D. R., Poock, G. K., and Richards, F. R., Experimentation Manual, Part I: Experimentation Methodology, (unpublished), Naval Postgraudate School, January 1980.

Digital Equipment Corporation, System Services Reference Manual, May 1982.

Digital Equipment Corporation, VAX-11 FORTRAN Language Reference Manual, April 1982.

Draper, N. R. and Smith, H., Applied Regression Analysis, 2d ed., Wiley, 1981.

Frey, P. W., Chess Skill in Man and Machine, Springer-Verlag, 1977.

Harkness, K., The Official Blue Book and Encyclopedia of Chess, David McKay Company, Inc., 1956.

Hicks, C. R., Fundamental Concepts in the Design of Experiments, 2d ed., Holt, Rinehart, and Winston, Inc., 1973.

Joint Chiefs of Staff, Joint War Gaming Agency, Joint War Gaming Manual, JWGA-167-69, July 1969.

Levy, D., Chess and Computers, Computer Science Press, 1976.

Newborn, M., Computer Chess, Academic Press, 1975.

Puscheck, Herbert C., The Development and Application of a Simple Wargame to the Study of Sequential Decision Making in a Conflict Environment, Ph.D. Thesis, Purdue University, 1969.

Shannon, C. E., "Programming a Computer to Play Chess", Scientific American, February 1950

Wonnacott, Thomas H. and Wonnacott, R. J., Introductory Statistics, 3d ed., Wiley, 1977.

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Technical Information Center Cameron Station Alexandria, Virginia 22314	2
2.	Library, Code 0142 Naval Postgraudate School Monterey, California 93943	2
3.	LtCol John T. Malokas, Code 39 Naval Postgraduate School Monterey, California 93943	2
4.	LCDR Kenneth M. O'Bryant 4813 Tinasa Way San Diego, California 92124	1
5.	CPT Robert G. Risney 530 N. 4th St. Missouri Valley, Iowa 51555	2
6.	Assoc Professor F. R. Richards, Code 55Rh Department of Operations Research Naval Postgraduate School Monterey, California 93943	1
7.	CDR Gary R. Porter, Code 55Pt Department of Operations Research Naval Postgraduate School Monterey, California 93943	1
8.	Dr. David Bash USA Nuclear and Chemical Agency ATTN: MONA-OPS (Dr. Bash) 7500 Backlich Road, Bldg. 2073 Springfield, Virginia 22150	1
9.	MAJ William Ted Farmer DEERS Support Office 2100 Garden Road, Suite #B-2 Monterey, California 93940	1
10.	CAPT Wayne P. Hughes (USN Ret), Code 55Hi Department of Operations Research Naval Postgraduate School Monterey, California 93943	1

11. Ted Yudachovski 1
Monterey Chess Center
430 Alvarado
Monterey, California 93940

END

FILMED

11-84

DTIC